

**Amendments to the Claims:**

1. (Currently Amended) A method comprising  
performing a ~~first type~~ checking on an object to ~~find out~~ detect a hotspot in the first type checking, and  
in response to detecting the hotspot, performing a first ~~second~~ type checking between a class of the object and a target class specified by the hotspot to assert an indicator in an object header of the object that is to indicate a success of the ~~second~~ first type checking at the hotspot in response to detecting the success.
2. (Canceled)
3. (Currently Amended) The method of claim 1 further comprising  
deasserting the indicator ~~to indicate in response to detecting a failure of the ~~second~~ first type checking at the hotspot between the object class and the target class.~~
4. (Currently Amended) The method of claim 1 further comprising  
skipping a ~~third~~ second type checking between the object class and the target class at the hotspot, in response to determining that the indicator of the object header is asserted.
5. (Currently Amended) The method of claim 1 further comprising  
skipping a ~~third~~ second type checking between the object class and the target class, in response to determining that the indicator of the object header is deasserted to indicate a failure of the ~~second~~ first type checking.
6. (Currently Amended) The method of claim 1 further comprising  
detecting the hotspot in the first ~~type~~ checking by dynamic profiling.
7. (Currently Amended) A system, comprising

a processor to ~~find-out~~ detect a hotspot in a first type checking for a class of an object, and ~~performing~~ perform a second type checking between the object class and a target class specified by the hotspot to indicate by an indicator in a header of the object a result of the second type checking at the hotspot; and  
a memory to save the target class.

8. (Currently Amended) The system of claim 7, wherein the processor further to  
determine whether a third type checking between the object class and the target class is successful at the hotspot based on an logic value of the indicator.

9. (Currently Amended) The system of claim 7, wherein the processor further to  
skipping a third type checking on the object at the hotspot, in response to detecting that the indicator has a second logic value that is to indicate a failure of the second type checking.

10. (Currently Amended) The system of claim 7, wherein the processor further to  
traverse a class hierarchy associated with the class of the object for the ~~second~~ first type chekcing.

11. (Previously Presented) The system of claim 7, wherein the processor further to  
assert the indicator in response to determining in the second type checking that the class of the object and the target class match a type checking condition.

12. (Previously Presented) The system of claim 7, wherein the processor further to  
return a signal indicating that the second type checking is successful, in response to determining that the class of the object and the target class match a predetermined criterion.

13. (Previously Presented) The system of claim 7, wherein the memory further to save a beginning address of a handle of the target class, and wherein the processor further to detect the hotspot by dynamic profiling.

14. (Currently Amended) A tangible machine readable storage medium comprising a plurality of instructions that in response to being executed result in a computing device

~~find out~~ detecting a hotspot in a first type checking for a class of an object , and performing a second type checking between the class of the object and a target class specified by the hotspot to indicate by a bit indicator in a header of the object a result of the second type checking at the hotspot.

15. (Previously Presented) The tangible machine readable storage medium of claim 14, wherein the machine readable medium further comprising instructions that in response to being executed result in the computing device

skipping a third type checking at the hotspot between the object class and the target class, in response to determining that the bit indicator is asserted to indicate a successful result of the second type checking.

16. (Previously Presented) The tangible machine readable storage medium of claim 14, wherein the machine readable medium further comprising instructions that in response to being executed result in the computing device

skipping a third type checking at the hotspot between the object class and the target class, in response to determining that the bit indicator is deasserted to indicate a failure result of the second type checking.

17. (Previously Presented) The tangible machine readable storage medium of claim 14, wherein the machine readable medium further comprising instructions that in response to being executed result in the computing device

detecting at the hotspot whether the object class and the target class match a type checking criterion based on the bit indicator.

18. (Previously Presented) The tangible machine readable storage medium of claim 16, wherein the machine readable medium further comprising instructions that in response to being executed result in the computing device

asserting the bit indicator, in response to determining in the second type checking that the object class and the target class match a type checking criterion.

19. (Previously Presented) The tangible machine readable storage medium of claim 16, wherein the machine readable medium further comprising instructions that in response to being executed result in the computing device

at the hotspot, returning a signal indicating whether the object class and the target class match a type checking criterion based on the bit indicator.

20. (Previously Presented) The tangible machine readable storage medium of claim 16, wherein the machine readable medium further comprising instructions that in response to being executed result in the computing device

throwing an exception at the hotspot, in response to determining from the bit indicator that the second type checking for the hotspot is failed.

21. (Previously Presented) The tangible machine readable storage medium of claim 16, wherein the machine readable medium further comprising instructions that in response to being executed result in the computing device

pushing a result code on a stack to indicate whether a type checking for the hotspot is successful based on the bit indicator.

22. (Currently Amended) A system comprising,

- a processor,
- a memory coupled to the processor, wherein the memory is to store
- a compiler to convert source code associated with a first time type checking for an object into byte code;
- a loader coupled with the compiler to load the byte code;

a dynamic compiler coupled with the loader to receive the byte code from the loader, and to generate first native code associated with the first time type checking based on the byte code; and

a profiler coupled with the dynamic compiler to detect a hotspot in the first time type checking based on the first native code, and to return the hotspot to the dynamic compiler,

wherein the dynamic compiler further to regenerate second native code that calls a second type checking for the hotspot to indicate a type checking result between a class of the object and a target class specified by the hotspot by an bit indicator in an object header of the object.

23. (Canceled)

24. (Currently Amended) The system of claim 22 ~~23~~, wherein the dynamic compiler further to assert an indicator in an object header of the object to a first logic value to indicate a success of the second type checking between a class of the object and a target class specified by the hotspot.

25. (Currently Amended) The system of claim 22 ~~23~~, wherein the dynamic compiler further to regenerate second native code that calls a second type checking for the hotspot to indicate a failure of the second type checking between a class of the object and a target class specified by the hotspot by a second logic value in a header of the object.

26. (Currently Amended) The system of claim 22 ~~23~~, wherein the dynamic compiler further to regenerate second native code that calls a type checking function for the hotspot to skip type checking between a class of the object and a target class specified by the hotspot, in response to determining that an object header of the object indicates a success of the second a type check.

27. (Currently Amended) The system of claim 22 ~~23~~, wherein the dynamic compiler further to regenerate second native code that calls a type checking function for the hotspot to skip type

checking between a class of the object and a target class specified by the hotspot, in response to determining that an indicator associated with the target class in an object header of the object is deasserted in the second type checking.

28. (Currently Amended) The system of claim 22 ~~23~~, wherein the dynamic compiler further to regenerate second native code that calls a type checking function for the hotspot to assert an indicator in an object header of the object, in response to a success of the second type checking ~~success~~ between a class of the object and a target class specified by the hotspot.

29. (Previously Presented) The system of claim 22, wherein the dynamic compiler further to regenerate second native code that calls a second type checking for the hotspot to traverse super classes of a class of the object to determine whether one of the super classes is the same as a target class as specified by the hotspot.

30. (Canceled)